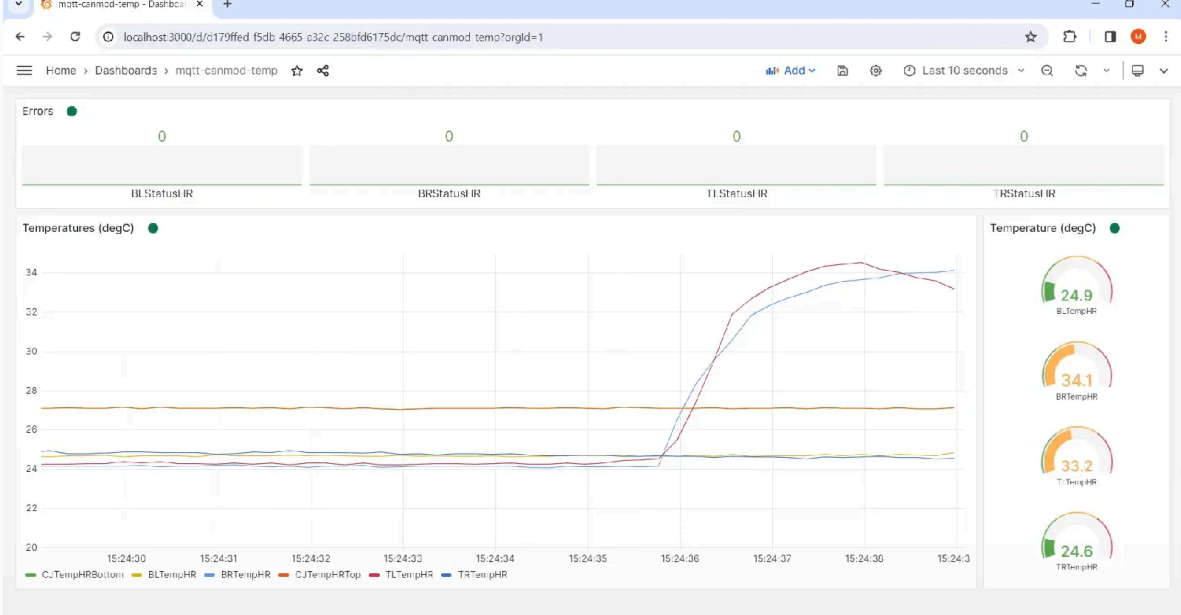


New Python API for CAN-USB streaming and TXT/MF4 log files

CSS Electronics newsletter - April 2024



Python CAN-USB streaming API [+Grafana-MQTT dashboards]

Need to stream CAN bus data via USB in Python?

With our new 100% free Python API you can connect the [CLX000](#) and/or [CANmod](#) via USB to your PC - and integrate the data with the popular python-can library!

KEY FEATURES:

- SIMPLE-TO-USE: Pip install, then stream data in <5 lines of code
- STREAM DATA: Send/receive raw CAN frames via CLX000/CANmod(s)
- DBC DECODE: Easily DBC decode data to physical values
- WINDOWS/LINUX: Deploy on Windows and Linux (incl. Raspberry)
- ECOSYSTEM: Leverage the extensive python-can extension tools
- API EXAMPLES: Benefit from excellent docs and script examples

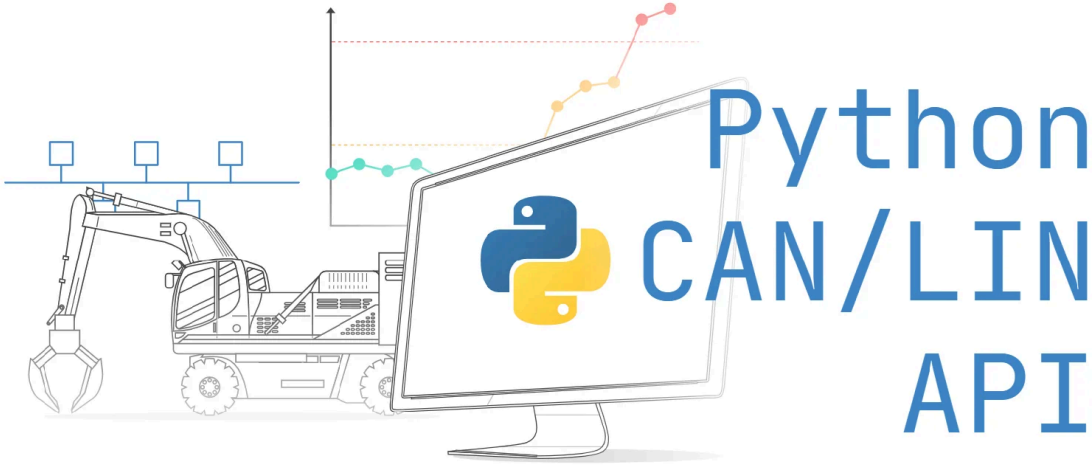
EXAMPLE USE CASES:

1. Deploy multiple CANmod.temp units in a temperature-sensitive production process to provide real-time alerts if temperatures exceed certain thresholds
2. Replay e.g. MF4/TXT log files entirely/selectively onto your CAN bus as part of an advanced CAN diagnostics/testing use case
3. Use our 'demo can2mqtt' example to create a [Grafana-MQTT integration](#) to display real-time CAN data in customizable Grafana dashboards!

We're excited to see what you deploy this for!

Get started with our simple-to-use script examples via the [CLX000 Intro](#) or [CANmod Intro](#) - or learn more in our article below:

[Learn more](#)



New Python MDF/TXT log file API

In an effort to further standardize the integration of log files across our CAN bus data loggers, we have now introduced two new Python API modules for log files:

1. [CANedge](#): python-can-csscan-mf4
2. [CLX000](#): python-can-csscan-txt

The modules let you directly load log files from the CANedge/CLX000 into the popular python-can library (like our new CAN-USB Python integration). This opens up access to a wide array of powerful tools for working with CAN bus data in Python.

One example is cantools, which adds support for DBC decoding data to physical values. Other tools enable protocol-specific support incl. J1939, CANopen, ISO-TP, UDS, XCP.

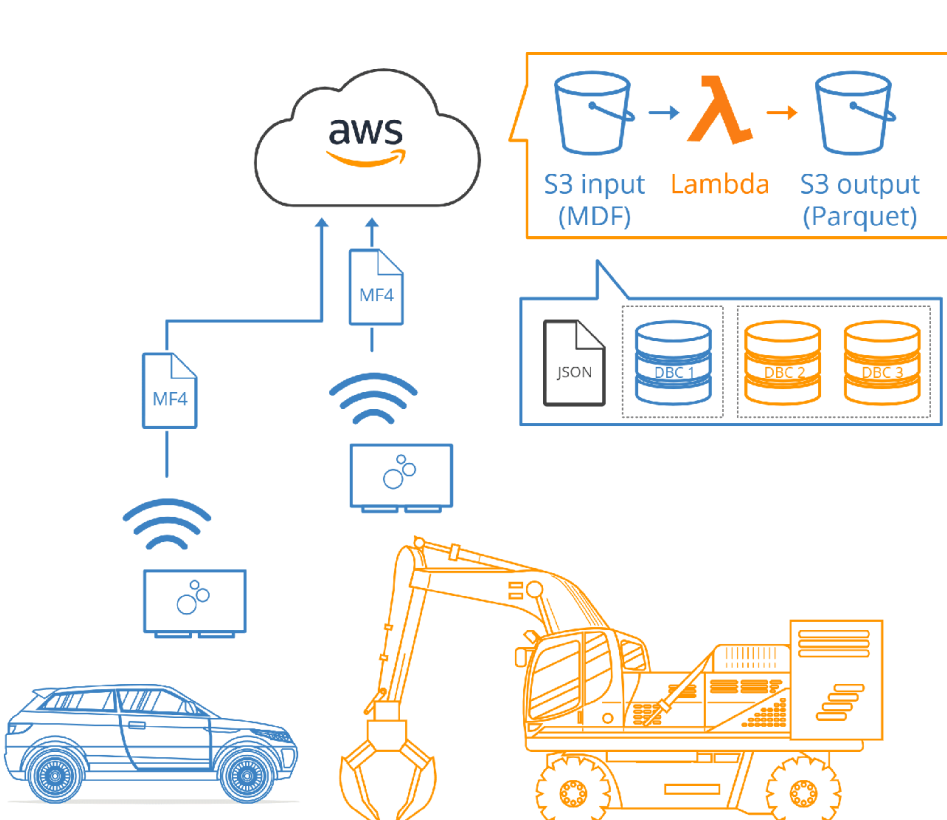
For the CANedge, this API replaces our 'legacy' Python API as explained [here](#).

As a result, you now have two options for working with your log files in Python:

1. If your goal is to analyse large amounts of DBC decoded CAN/LIN data (across many log files), we recommend using the **MF4/TXT decoders to create a Parquet data lake**. This will make it easier to 'abstract away' the log file aspect of your data and enables 'beyond-memory' analyses. The Parquet data lake can also serve other purposes, such as Grafana dashboard visualization
2. For more specialized use cases, the new Python API enables **direct loading of the log files into python-can**. This is useful if you wish to analyse the raw CAN/LIN frames, replay the data through a USB-to-CAN interface, selectively DBC decode messages/signals, inject conditional logic etc

Get started with our simple-to-use script examples via the [CANedge Intro](#) or [CLX000 Intro](#) - or learn more in our article below:

[Learn more](#)



AWS S3 Parquet data lakes + device specific DBC files

The CANedge lets you log CAN/LIN data to an SD card - and automatically upload the data to your own S3 bucket. Here you can easily deploy automation to create an [AWS S3 based Parquet data lake](#). Simply store your DBC files in your S3 input bucket and deploy our Lambda function to automatically DBC decode incoming log files.

The data lake is e.g. useful in setting up [Grafana-Athena dashboards](#).

In our latest update, you can now also add a 'dbc-groups.json' file in your S3 input bucket. In this you can specify which DBC files you want to apply across which devices.

This is useful in managing S3 buckets that contain different vehicle/machinery models or types - in particular if these have separate DBC files with conflicting CAN IDs.

[Learn more](#)

Status at CSS

Our current focus is on finalizing development of an upcoming new CANmod product, as well as the next CANedge firmware update. Stay tuned for news on both!